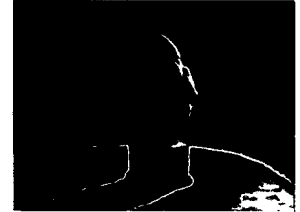# EXHIBIT K

Subsurface scattering

From Wikipedia, the free encyclopedia

Subsurface scattering (or SSS) is a mechanism of light transport in which light penetrates the surface of a translucent object, is scattered by interacting with the material, and exits the surface at a different point. The light will generally penetrate the surface and be reflected a number of times at irregular angles inside the material, before passing back out of the material at an angle other than the angle it would reflect at had it reflected directly off the surface. Subsurface scattering is important in 3D computer graphics, being necessary for the realistic rendering of materials such as marble, skin, and milk.

Three dimensional object with subsurface scattering

Contents

- 1 Rendering Techniques
    - 1.1 Depth Map based SSS
    - 1.2 Texture Space Diffusion
- 2 External links
- 3 References

Rendering Techniques

Most materials used in real-time computer graphics today only account for the interaction of light at the surface of an object. In reality, many materials are slightly translucent: light enters the surface; is absorbed, scattered and re-emitted - potentially at a different point.

Skin is a good case in point; only about 6% of reflectance is direct, 94% is from subsurface scattering[1]. An inherent property of translucent materials is absorption. The further through the material light travels, the greater the proportion absorbed. To simulate this effect, a measure of the distance light has traveled through the material must be obtained.

Depth Map based SSS

One method of estimating this distance is to use depth maps[2], in a manner similar to shadow mapping. The scene is rendered from the light's point of view into a depth map, so that the distance to the nearest surface is stored. The depth map is then projected onto it using standard projective texture mapping and the scene re-rendered. In this pass, when shading a given point, the distance from the light at the point the ray entered the surface can be obtained by a simple texture lookup. By subtracting this value from the point the ray exited the object we can gather an estimate of the distance the light has traveled through the object.

Depth estimation using depth maps

The measure of distance obtained by this method can be used in several ways. One such way is to use it to index directly into an artist created 1D texture that falls off exponentially with distance. This approach, combined with other more traditional lighting models, allows the creation of different materials such as marble, jade and wax.

Potentially, problems can arise if models are not convex, but depth peeling[3] can be used to avoid the issue. Similarly, depth peeling can be used to account for varying densities beneath the surface, such as bone or muscle, to give a more accurate scattering model.

As can be seen in the image of the wax head to the right, light isn't diffused when passing through object using this technique; back features are clearly shown. One solution to this is to take multiple samples at different points on surface of the depth map. Alternatively, a different approach to approximation can be used, known as texture-space diffusion.

Texture Space Diffusion

As noted at the start of the section, one of the more obvious effects of subsurface scattering is a general blurring of the diffuse lighting. Rather than arbitrarily modifying the diffuse function, diffusion can be more accurately modeled by simulating it in texture space. This technique was pioneered in rendering faces in The Matrix Reloaded[4], but has recently fallen into the realm of real-time techniques.

The method unwraps the mesh of an object using a vertex shader, first calculating the lighting based on the original vertex coordinates. The vertices are then remapped using the UV texture coordinates as the screen position of the vertex, suitable transformed from the [0, 1] range of texture coordinates to the [-1, 1] range of normalized device coordinates. By lighting the unwrapped mesh in this manner, we obtain a 2D image representing the lighting on the object, which can then be processed and reapplied to the model as a normal map. To simulate diffusion, the light map texture can simply be blurred. Rendering the lighting to a lower-resolution texture in itself provides a certain amount of blurring. The amount of blurring required to accurately model subsurface scattering in skin is still under active research, but performing only a single blur poorly models the true effects[5]. To emulate the wavelength dependent nature of diffusion, the samples used during the (Gaussian) blur can be weighted by channel. This is somewhat of an artistic process. For human skin, the broadest scattering is in red, then green, and blue has very little scattering.

A major benefit of this method is its independence of screen resolution; shading is performed only once per texel in the texture map, rather than for every pixel on the object. An obvious requirement is thus that the object have a good UV mapping, in that each point on the texture must map to only one point of the object. Additionally, the use of texture space diffusion causes implicit soft shadows, alleviating one of the more unrealistic aspects of standard shadow mapping.

External links

- Henrik Wann Jensen's subsurface scattering website (http://graphics.ucsd.edu/~henrik/images/subsurf.html)
- An academic paper by Jensen on modeling subsurface scattering (http://graphics.ucsd.edu/~henrik/papers/bssrdf/)
- Maya Tutorial - Subsurface Scattering: Using the Misss_Fast_Simple_Maya shader (http://www.highend3d.com/maya/tutorials/rendering_lighting/shaders/135.html)

References

1. ^ Krishnaswamy, A (2004). 'A Biophysically-based Spectral Model of Light Interaction with Human Skin (http://eg04.inrialpes.fr/Programme/Papers/PDF/paper1189.pdf. *Computer Graphics Forum* 23 (3).
2. ^ Green, Simon (2004). "Real-time Approximations to Subsurface Scattering". *GPU Gems*: 263-278.
3. ^ Nagy, Z (2003). 'Depth-Peeling for Texture-based Volume Rendering (http://cg.cs.uni-bonn.de/docs/publications/2003/nagy-2003-depth.pdf
". *11th Pacific Conference on Computer Graphics and Applications*. 429.
4. ^ Borshukov, G (2005). 'Realistic human face rendering for "The Matrix Reloaded" (http://www.virtualcinematography.org/publications/acrobat/Face-s2 ". *Computer Graphics*.
5. ^ d'Eon, E (2007). "Advanced Skin Rendering (http://developer.download.nvidia.com/presentations/2007/gdc/Adva ". *GDC 2007*.

Retrieved from "http://en.wikipedia.org/wiki/Subsurface_scattering"

Categories: Computer stubs | 3D computer graphics | Optics